UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/982,395	10/18/2001	Arild E. Skjolsvold	MS1-2624US	7192

22801 7590 03/15/2006

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER
BONSHOCK, DENNIS G

ART UNIT	PAPER NUMBER
2173	

DATE MAILED: 03/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.**	**Applicant(s)**
	09/982,395	SKJOLSVOLD, ARILD E.
	Examiner	**Art Unit**
	Dennis G. Bonshock	2173

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 December 2005*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-11,13-15,17-28 and 34-39* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-11, 13-15, 17-28, and 34-39* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)
2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) □ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) □ Notice of Informal Patent Application (PTO-152)
6) □ Other: _____.

Final Rejection

Response to Amendment

1. It is hereby acknowledged that the following papers have been received and placed on record in the file: Amendment as received on 12-28-2005.

2. Claims 1-39 have been examined.

Status of Claims:

3. Claims 1-11, 13-15, 17-28, and 34-39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Parker et al., Patent #5,781,720, hereinafter Parker and Singh el al., Patent #6,415,396, hereinafter Singh.

4. Claims 12, 16, and 29-33 have been cancelled by the applicant.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-11, 13-15, 17-28, and 34-39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Parker et al., Patent #5,781,720, hereinafter Parker and Singh el al., Patent #6,415,396, hereinafter Singh.

3. With regard to claim 1, Parker teaches a system that does automated testing of a GUI environment, through the generation of a mapping between GUI objects and their functions (see column 4, lines 1-26, column 16, line 53 through column 17, line 12,

column 25, lines 4-8 and column 9, lines 50-67), the executing of an executable feature of the Logical Screen Element (LSE) (see column 4, lines 39-45), a LSE Manager that identifies locations of the LSEs (see column 10, line 1-9), and storing the information for GUI objects in tables in the GUI and in the memory (see column 12, lines 50-65, column 4, lines 39-45, and column 9, lines 11-21). Parker further teaches, in column 9, lines 50-67, the LSEM storing functions that correspond to (are mapped to) objects on the screen, and in column 12, lines 50-56, the test driver having access to the LSEM for driving the application. In summary, Parker's system generates tests by using a test outline that does not have any GUI specific references, during execution of the test, the test tool asks the GUI about the object under tests location and state and gives the object a logical name by giving it a mapping, just before implementation of the actual test on the object (see column 16, line 53 through column 17, line 12). Parker, however, doesn't explicitly state updating the association in the map data structure upon execution of the executable feature. Singh teaches a system that automatically generates test sets and specifically uses regression tests (see column 3, lines 35-60), similar to that of Parker, but further explicitly points out selection techniques (updating the state in the traversal through a group of elements) used on a GUI to provide navigation through a graphical test structure (see column 3, lines 25-59, column 11, lines 13-30, and column 13, line 50 through column 14, line 9 and in figure 6). It would have been obvious to one of ordinary skill in the art, having the teachings of Parker and Singh before him at the time the invention was made to modify the system of Parker to include the selection techniques of Singh. One would have been motivated to make such a combination because Parker and Singh both automatically generate test sets and implement regression testing, they only choose to do selection in different manners.

4. With regard to claims 2, and 23, which teach a system in which selection of an executable feature exposes a second graphic feature that is then treated the same as the first, Parker teaches, in column 4, lines 50-55 and column 9, lines 9-22, that when one element exposes another element the second element is processed likewise. Singh further teaches this limitation, in column 4, lines 58-60, which teach that if a given requirement is executed, and a new event is received, the second requirement is executed.

5. With regard to claims 3 and 24, which teach the retrieving comprising capturing information pertaining to the graphic element, Parker teaches, in column 30, lines 15-19, a comparison based on captured information.

6. With regard to claims 4, 21, 25, and 36, which teach that storing includes updating an indicator associated with the graphics element when an executable feature stored in association with the graphics element is executed, Parker further teaches, in column 27, lines 60-65, graphical items having a Boolean value to show if the item is currently executable. It would be obvious having the teachings of Parker and Singh that the Boolean value of Parker could be controlled to only execute each item once similar to the systematic selection techniques (depth-first/breadth-first) of Singh.

7. With regard to claim 5, which teaches storing including organizing the retrieved information so that an executable feature stored in association with graphics element can be interpreted by a computer-executable application capable of accessing the

retrieved information, Parker teaches, in column 12, lines 50-65, the test driver accessing the information stored in accordance with the graphical objects.

8. With regard to claim 6, which teaches storing including organizing the retrieved information such that an executable feature stored in association with the graphics element can be interpreted by a user capable of accessing the retrieved information from memory, Parker teaches, in column 12, lines 50-65, a GUI that stores all information needed for the GUI objects in tables within the GUI and in memory.

9. With regard to claim 7, which teaches selecting the executable feature based on the association stored in the map data structure, Parker teaches, in column 9, lines 50-67, executable features (function) in the LSM, having corresponding user-visible elements on the screen. Parker further teaches, in column 12, lines 50-65, a GUI that stores all information needed for the GUI objects in tables within the GUI and in memory.

10. With regard to claims 8 and 17, which teach selecting comprising (deterministically) selecting an executable feature not previously executed, Parker further teaches, in column 27, lines 60-65, graphical items having a Boolean value to show if the item is currently executable. It would be obvious having the teachings of Parker and Singh that selection could be made with respect to the Boolean value of Parker that could be controlled to only execute each item only once, similar to the systematic selection techniques (depth-first/breadth-first) of Singh.

11. With regard to claims 9, 18, and 26, which teach the selecting comprising reviewing an indicator to select an executable feature not previously executed, Parker

further teaches, in column 27, lines 60-65, graphical items having a Boolean value to show if the item is currently executable. It would be obvious having the teachings of Parker and Singh that selection could be made with respect to the Boolean value of Parker that could be controlled to only execute each item only once similar to the selection techniques (depth-first/breadth-first) of Singh.

12. With regard to claims 10, 19, 27, and 39, which teach selecting comprising (deterministically) selecting executable features in a depth-first mode of operation, Singh further teaches, in column 3, lines 47-51 and column 13, lines 50-63, reaching nodes in a hierarchical graph through the use of selection techniques applied to the graph to generate test cases, and specifically pointed out traversing a graph in a depth-first manner.

13. With regard to claims 11, 20, 28, and 38, which teach selecting comprising (deterministically) selecting executable features in a breadth-first mode of operation, Singh further teaches, in column 3, lines 47-51 and column 13, lines 50-63, reaching nodes in a hierarchical graph through the use of selection techniques applied to the graph to generate test cases, and specifically pointed out traversing a graph in a breadth-first manner.

14. With regard to claim 13, Parker teaches a system that does automated testing of a GUI environment, through the generation of a mapping between GUI objects and their functions (see column 4, lines 1-26, column 16, line 53 through column 17, line 12, column 25, lines 4-8 and column 9, lines 50-67), a comparison based on captured information (see column 30, lines 15-19), the executing of an executable feature of the

Logical Screen Element (LSE) (see column 4, lines 39-45), a LSE Manager that identifies locations of the LSEs (see column 10, line 1-9), and storing the information for GUI objects in tables in the GUI and in the memory (see column 12, lines 50-65, column 4, lines 39-45, and column 9, lines 11-21). Parker further teaches, in column 9, lines 50-67, the LSEM storing functions that correspond to (are mapped to) objects on the screen, and in column 12, lines 50-56, the test driver having access to the LSEM for driving the application. In summary, Parker's system generates tests by using a test outline that does not have any GUI specific references, during execution of the test, the test tool asks the GUI about the object under tests location and state and gives the object a logical name by giving it a mapping, just before implementation of the actual test on the object (see column 16, line 53 through column 17, line 12). Parker, however, doesn't explicitly state deterministically selecting one of the executable features stored in the map data structure. Singh teaches a system that automatically generates test sets and specifically uses regression tests (see column 3, lines 35-60), similar to that of Parker, but further explicitly points out selection techniques (systematically updating the state in the traversal through a group of elements) used on a GUI to provide navigation through a graphical test structure (see column 3, lines 25-59, column 11, lines 13-30, and column 13, line 50 through column 14, line 9 and in figure 6). It would have been obvious to one of ordinary skill in the art, having the teachings of Parker and Singh before him at the time the invention was made to modify the system of Parker to include the selection techniques of Singh. One would have been motivated to make such a combination because Parker and Singh both automatically generate test sets and implement regression testing, they only choose to do selection in different manners.

15. With regard to claim 14, which teaches the capture agent being invoked by the application driver, Parker further teaches, in column 4, lines 15-20 and in column 30, lines 15-19, a comparison based on captured information executed by a test driver on the application program.

16. With regard to claim 15, which teaches the capture agent submitting retrieved information to the application driver, Parker further teaches, in column 4, lines 15-20 and in column 30, lines 15-19, a comparison based on captured information executed by a test driver on the application program.

17. With regard to claim 22, Parker teaches a system that does automated testing of a GUI environment, through the generation of a mapping between GUI objects and their functions (see column 4, lines 1-26, column 16, line 53 through column 17, line 12, column 25, lines 4-8 and column 9, lines 50-67), the executing of an executable feature of the Logical Screen Element (LSE) (see column 4, lines 39-45), a LSE Manager that identifies locations of the LSEs (see column 10, line 1-9), and storing the information for GUI objects in tables in the GUI and in the memory (see column 12, lines 50-65, column 4, lines 39-45, and column 9, lines 11-21). Parker further teaches, in column 9, lines 50-67, the LSEM storing functions that correspond to (are mapped to) objects on the screen, and in column 12, lines 50-56, the test driver having access to the LSEM for driving the application. In summary, Parker's system generates tests by using a test outline that does not have any GUI specific references, during execution of the test, the

test tool asks the GUI about the object under tests location and state and gives the object a logical name by giving it a mapping, just before implementation of the actual test on the object (see column 16, line 53 through column 17, line 12). Parker further teaches graphical items having a Boolean value to show if the item is currently executable (see column 27, lines 60-65) but doesn't explicitly state selecting one of the executable features that has not been previously executed. Singh teaches a system that automatically generates test sets and specifically uses regression tests (see column 3, lines 35-60), similar to that of Parker, but further explicitly points out selection techniques (systematically updating the state in the traversal through a group of elements) used on a GUI to provide navigation through a graphical test structure to provide optimal use of time (see column 3, lines 25-59, column 11, lines 13-30, and column 13, line 50 through column 14, line 9 and in figure 6), specifying depth-first and breadth-first navigation which is known in the art to navigate to each of the extremes only once. It would be obvious having the teachings of Parker and Singh that selection could be made with respect to the Boolean value of Parker that could be controlled to only execute each item only once, similar to the systematic selection techniques (depth-first/breadth-first) of Singh; and to modify the system of Parker to include the selection techniques of Singh. One would have been motivated to make such a combination because Parker and Singh both automatically generate test sets and implement regression testing, they only choose to do selection in different manners.

18. With regard to claim 34, Parker teaches a system that does automated testing of a GUI environment, through the generation of a mapping between GUI objects and their

functions (see column 4, lines 1-26, column 16, line 53 through column 17, line 12, column 25, lines 4-8 and column 9, lines 50-67), determining the state of the GUI through a comparison based on captured information (see abstract and column 30, lines 15-19), the executing of an executable feature of the Logical Screen Element (LSE) (see column 4, lines 39-45), a LSE Manager that identifies locations of the LSEs (see column 10, line 1-9), and storing the information for GUI objects in tables in the GUI and in the memory (see column 12, lines 50-65, column 4, lines 39-45, and column 9, lines 11-21). Parker further teaches, in column 9, lines 50-67, the LSEM storing functions that correspond to (are mapped to) objects on the screen, and in column 12, lines 50-56, the test driver having access to the LSEM for driving the application. In summary, Parker's system generates tests by using a test outline that does not have any GUI specific references, during execution of the test, the test tool asks the GUI about the object under tests location and state and gives the object a logical name by giving it a mapping, just before implementation of the actual test on the object (see column 16, line 53 through column 17, line 12). Parker, however, doesn't explicitly state deterministically selecting one of the executable features stored in the map data structure and repeating for a new element. Singh teaches a system that automatically generates test sets and specifically uses regression tests (see column 3, lines 35-60), similar to that of Parker, but further explicitly points out selection techniques (systematically updating the state in the traversal through a group of elements) used on a GUI to provide navigation through a graphical test structure (see column 3, lines 25-59, column 11, lines 13-30, and column 13, line 50 through column 14, line 9 and in figure 6) specifying depth-first and breadth-first navigation which is known in the art to navigate to each of the extreme once. It would have been obvious to one of ordinary skill in the art, having the teachings of Parker and Singh before him at the time the invention was made to modify the system of Parker to include the selection techniques of Singh. One would have been motivated to make such a combination because Parker and Singh both automatically generate test sets and implement regression testing, they only choose to do selection in different manners.

19. With regard to claim 35, which teaches storing the association between each graphics element with its corresponding executable feature in a map, Parker teaches a system that does automated testing of a GUI environment, through the generation of a mapping between GUI objects and their functions (see column 4, lines 1-26, column 16, line 53 through column 17, line 12, column 25, lines 4-8 and column 9, lines 50-67), and storing the information for GUI objects in tables in the GUI and in the memory (see column 12, lines 50-65, column 4, lines 39-45, and column 9, lines 11-21).

20. With regard to claim 37, which teaches deterministically selecting one of the graphics elements uses a map, Singh teaches selection techniques (systematically updating the state in the traversal through a group of elements) used on a GUI to provide navigation through a graphical test structure (see column 3, lines 25-59, column 11, lines 13-30, and column 13, line 50 through column 14, line 9 and in figure 6) specifying both depth-first and breadth-first navigation.

Response to Arguments

21. The arguments filed on 12-28-2005 have been fully considered but they are not persuasive. Reasons set forth below.

22. The applicants' argue that there is no suggestion or motivation to combine the references.

23. In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, the examiner respectfully submits that both teach systems that automatically generates test sets and specify regression tests, they only choose to do selection in different manners. Singh further teaches the testing of features that would be visually displayed to the user in an ATM (as in column 6, lines 21-37), similar to the testing of a GUI of Parker.

24. The applicants' argue that the testing script according to the Parker reference is not dynamic during the testing process. "It is produced before testing occurs; if changes are required, they must be made by a human and while testing is not occurring."

25. In response, the examiner respectfully submits that Parker teaches, in column 4, lines 1-26 and in column 16, line 53 through column 17, line 12, the system using a script, but the script not being directed to GUI specific references (generic test script),

the "test executive" automatically creates these GUI specific references during execution. This system, much like the one claimed, has no associations with GUI specific elements until runtime. At runtime, just before making use of the information to implement the test (initiating a click), a GUI object is given a logical name, in the mapping, for the purposes of testing.

26. The applicants' argue that the claims of the present invention recite that an association is stored "during execution of the software being tested."

27. In response, the examiner respectfully submits that the claims actually recite that "the graphics element (is) rendered during execution of the software being tested." As recited above, and column 16, line 53 through column 17, line 12, Mappings are created during execution, using the generic test script supplemented by logical names associated with screen elements.

Conclusion

28. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

29. A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

30. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dennis G. Bonshock whose telephone number is (571) 272-4047. The examiner can normally be reached on Monday - Friday, 6:30 a.m. - 4:00 p.m.

31. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cabeca can be reached on (571) 272-4048. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

32. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

3-13-06
dgb

RAYMOND J. BAYERL
PRIMARY EXAMINER
ART UNIT 2173